

NUMERICAL APPROXIMATION OF INTERNAL DISCONTINUITY INTERFACE PROBLEMS*

MARCO DISCACCIATI[†], ALFIO QUARTERONI[‡], AND SAMUEL QUINODOZ[§]

Abstract. This work focuses on the finite element discretization of boundary value problems whose solution features either a discontinuity or a discontinuous conormal derivative across an interface inside the computational domain. The interface is characterized via a level set function. The discontinuities are accounted for by using suitable extension operators whose numerical implementation requires a very low computational effort. After carrying out the error analysis, numerical results to validate our approach are presented in one, two, and three dimensions.

Key words. finite elements, interface problem, level set

AMS subject classifications. 65N12, 65N30

DOI. 10.1137/110850487

1. Introduction. This paper is focused on the numerical approximation of elliptic problems whose solution features discontinuities across interfaces internal to the computational domain. We consider a Poisson problem in two disjoint subdomains of the computational domain $\Omega \subset \mathbb{R}^N$ ($N = 1, 2, 3$) with jump conditions across the interface Γ separating the two subregions. Γ is a point if $N = 1$, a line if $N = 2$, or a surface if $N = 3$ that is characterized by a level set function $\phi : \Omega \rightarrow \mathbb{R}$. We consider an open bounded domain Ω partitioned into the two nonoverlapping subdomains $\Omega_1 = \{x \in \Omega \mid \phi(x) < 0\}$ and $\Omega_2 = \Omega \setminus \Omega_1 = \{x \in \Omega \mid \phi(x) > 0\}$. See Figure 1.1 for two possible instances. According to the classical level set method (see, e.g., [26] and [33]), ϕ is regarded as the signed distance function to the interface, whence Γ is defined by the equation $\phi = 0$.

The mathematical formulation of our problem is as follows. We look for a function u in Ω that satisfies a Poisson problem in each subdomain:

$$(1.1) \quad -\Delta u_i = f_i \quad \text{in } \Omega_i, \quad i = 1, 2,$$

where $u_i = u|_{\Omega_i}$, with the following conditions on the jumps of the trace and of the normal derivatives across Γ :

$$(1.2) \quad \llbracket u \rrbracket_\Gamma = u_2|_\Gamma - u_1|_\Gamma = g_d,$$

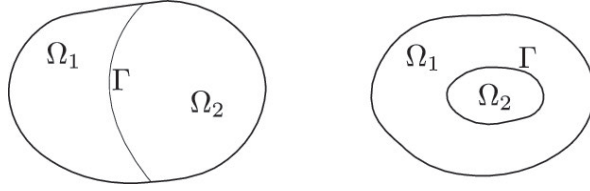
$$(1.3) \quad \left\llbracket \frac{\partial u}{\partial n} \right\rrbracket_\Gamma = \nabla u_2 \cdot n_2|_\Gamma + \nabla u_1 \cdot n_1|_\Gamma = g_n.$$

*Submitted to the journal's Methods and Algorithms for Scientific Computing section October 6, 2011; accepted for publication (in revised form) August 19, 2013; published electronically October 22, 2013. This work was supported by the Swiss National Science Foundation, Project Sinergia 125444: "Fluid dynamics and mixing behavior in orbitally shaken bioreactors for mammalian cell cultivation." <http://www.siam.org/journals/sisc/35-5/85048.html>

[†]Laboratori de Càlcul Numèric (LaCàN), Departament de Matemàtica Aplicada III (MA3), Universitat Politècnica de Catalunya (UPC BarcelonaTech), Campus Nord UPC - C2, E-08034 Barcelona, Spain (marco.discacciati@upc.edu).

[‡]MATHICSE, Chair of Modeling and Scientific Computing, Ecole Polytechnique Fédérale de Lausanne, Station 8, CH-1015 Lausanne, Switzerland. MOX, Politecnico di Milano, P.zza Leonardo da Vinci 32, I-20133 Milano, Italy (alfio.quarteroni@epfl.ch).

[§]MATHICSE, Chair of Modeling and Scientific Computing, Ecole Polytechnique Fédérale de Lausanne, Station 8, CH-1015 Lausanne, Switzerland (samuel.quinodoz@epfl.ch).

FIG. 1.1. Two examples of partition of the domain Ω .

g_d and g_n are two assigned functions on Γ , while n_1 and n_2 are the unit normal vectors on Γ directed outwards of Ω_1 and Ω_2 , respectively. Notice that $n_2 = -n_1$ on Γ . For simplicity, we impose homogeneous Dirichlet boundary conditions on the boundary of the domain Ω : $u = 0$ on $\partial\Omega$. We assume $f_i \in L^2(\Omega_i)$, $g_n \in L^2(\Gamma)$, and either $g_d \in H^{1/2}(\Gamma)$ if $\partial\Omega \cap \Gamma = \emptyset$ or $g_d \in H_{00}^{1/2}(\Gamma)$ if $\partial\Omega \cap \Gamma \neq \emptyset$ (we refer to [22] for a rigorous definition of these trace spaces).

Discontinuities in the normal derivative of the solution of a PDE arise, e.g., as soon as a force is localized on a part of the computational domain with lower dimension. The most popular example is the surface tension in flow simulations [27], a force that applies at the interface between two fluids. In this framework, Γ is a free surface, that is its location is a further unknown of the problem at hand. It can also happen that the solution itself is discontinuous, for example, with a phase transition when considering entropy [7] or with incompressible flames [25].

A possible strategy to correctly approximate this kind of problem is to build a mesh which captures the interface. This approach however may not be convenient. For instance, as for a time dependent problem, conforming meshes (those that perfectly match on the interface) have to be rebuilt at each time step, resulting in too expensive schemes in terms of computational cost. The mesh could also be cut by the interface and only locally rebuilt, but this could lead to highly deformed cells. Methods that do not require the reconstruction of a new mesh are in general preferable. For the same reasons, a good method should neither require mesh refinement near the interface, nor the computation of quantities (such as integrals) on the interface itself.

A method for solving problem (1.1) with a nonconforming mesh cannot rely on standard finite element discretization. Indeed, it has been shown in [17] that if a Lagrangian finite element space is used to approximate the solution, then one cannot expect a convergence rate better than \sqrt{h} (h being the finite element grid size). This has been the main motivation for the development of new methods, such as the extended finite element method (XFEM) [4] and a variant of Nitsche's method [18]. These methods share the same strategy: the finite element space is enriched with additional shape functions especially designed to resolve the discontinuities across the interface Γ . Extensions, e.g., to high order [8] or to open interfaces [35], can be realized by changing the enrichment functions and, if necessary, modifying the formulation. Even though they have been successfully applied to different kinds of multiphysics problems (e.g., [1, 9, 17]), still they are affected by some inherent weaknesses. The addition of these new shape functions can worsen the conditioning of the algebraic system and preconditioning strategies have to be redesigned for these methods (see, e.g., [2]). Moreover, the parallel implementation of such methods might be troublesome when dealing with time dependent problems, as the number of degrees of freedom might vary while the interface is evolving.

The goal of this paper is to present a new method, called SESIC (standing for simplified exact subgrid interface correction), that takes into account these jump

conditions in a finite element framework. Our method has been inspired by the ESIC method developed in [20]: we have improved the construction of the liftings and changed the weak formulation so that the efficiency of ESIC is maintained while improving its mathematical interpretation and its effective implementation. Indeed, in the SESIC method no polynomial refinement is required in the interface zone to obtain optimal convergence and, under certain conditions, the interface does not need to be explicitly reconstructed. The paper has the following content. In section 2, we study the weak formulation of the internal discontinuity problem (1.1)–(1.3) and we introduce at the continuous level suitable lifting operators to account for the jumps across the interface. In section 3, we present the finite element approximation of the problem and discuss several numerical properties of our approach in comparison with the one developed in [20]. In section 4 we provide the error analysis of our method. Finally, in sections 5 and 6 we illustrate the numerical results that we have obtained on different test cases.

2. Weak formulation for the internal discontinuity interface problem.

To derive a weak formulation of (1.1)–(1.3), we introduce two suitable liftings (also called extensions) $R_i g_d$ ($i = 1, 2$) of g_d in Ω_i so that the jump of $R_i g_d$ is g_d on Γ : $R_i g_d \in H^1_{\partial\Omega_i \setminus \Gamma}(\Omega_i) = \{v \in H^1(\Omega_i) | v = 0 \text{ on } \partial\Omega_i \setminus \Gamma\}$ ($i = 1, 2$) such that $\llbracket R_i g_d \rrbracket_\Gamma = g_d$. The trace theorem (see [23]) guarantees that such a lifting operator exists. Then, we consider the splitting

$$(2.1) \quad u_i = \bar{u}_i + R_i g_d \quad \text{in } \Omega_i.$$

We denote $\bar{u} : \Omega \rightarrow \mathbb{R}$ such that $\bar{u}_i = \bar{u}|_{\Omega_i}$. The function \bar{u} belongs to $H^1_0(\Omega)$. We consider a global test function $v \in H^1_0(\Omega)$ and its restrictions v_i on Ω_i . Then, on each domain, starting from (1.1), integrating by parts and exploiting the homogeneous Dirichlet boundary conditions on $\partial\Omega_i \cap \partial\Omega$, we obtain

$$(2.2) \quad \int_{\Omega_i} \nabla u_i \cdot \nabla v_i - \int_{\Gamma} \frac{\partial u_i}{\partial n_i} v_i = \int_{\Omega_i} f_i v_i, \quad i = 1, 2.$$

Summing up the contributions of each subdomain and imposing the jump condition on the normal derivative (1.3) in a natural way, we obtain

$$(2.3) \quad \sum_{i=1}^2 \int_{\Omega_i} \nabla u_i \cdot \nabla v_i - \int_{\Gamma} g_n v_i = \sum_{i=1}^2 \int_{\Omega_i} f_i v_i.$$

Finally, using the decomposition (2.1), we obtain the weak form of problem (1.1)–(1.3): find $\bar{u} \in H^1_0(\Omega)$ such that $\forall v \in H^1_0(\Omega)$

$$(2.4) \quad \int_{\Omega} \nabla \bar{u} \cdot \nabla v = \sum_{i=1}^2 \int_{\Omega_i} f_i v_i - \sum_{i=1}^2 \int_{\Omega_i} \nabla R_i g_d \cdot \nabla v_i + \int_{\Gamma} g_n v.$$

An alternative weak formulation of (1.1)–(1.3) was proposed by Huh and Sethian [20], by considering an additional lifting for the function g_n . More precisely, they define a function $S_i g_n \in H^1_{\partial\Omega_i \setminus \Gamma}(\Omega_i)$, $i = 1, 2$, such that $\llbracket \frac{\partial S_i g_n}{\partial n} \rrbracket_\Gamma = g_n$ and, instead of (2.1), they consider the three-term splitting

$$(2.5) \quad u_i = \hat{u}_i + R_i g_d + S_i g_n.$$

Since we have assumed that $g_n \in L^2(\Gamma)$, the existence of $S_i g_n \in H^1_{\partial\Omega_i \setminus \Gamma}(\Omega_i)$ is ensured by the solution of the Neumann problem for the Laplace equation. The splitting (2.5) is similar to the one proposed in [3] in a finite difference context. The two liftings $R_i g_d$ and $S_i g_n$ should ideally satisfy the following constraints:

$$(2.6) \quad \llbracket R_i g_d \rrbracket_\Gamma = g_d, \quad \left\llbracket \frac{\partial R_i g_d}{\partial n} \right\rrbracket_\Gamma = 0,$$

$$(2.7) \quad \llbracket S_i g_n \rrbracket_\Gamma = 0, \quad \left\llbracket \frac{\partial S_i g_n}{\partial n} \right\rrbracket_\Gamma = g_n,$$

in which case they would take into account the jump of the functions and that of the fluxes independently.

Using these lifting operators, the following weak form of problem (1.1)–(1.3) can be derived: find $\hat{u} \in H^1_0(\Omega)$ such that, $\forall v \in H^1_0(\Omega)$,

$$(2.8) \quad \int_\Omega \nabla \hat{u} \cdot \nabla v = \sum_{i=1}^2 \int_{\Omega_i} f_i v_i - \sum_{i=1}^2 \int_{\Omega_i} \nabla(R_i g_d + S_i g_n) \cdot \nabla v_i + \int_\Gamma g_n v.$$

Note that \hat{u} is such that $\hat{u}|_{\Omega_i} = \hat{u}_i$ for $i = 1, 2$.

We can remark that the bilinear form associated with both methods (2.4) and (2.8) is the classical Dirichlet formulation of the Poisson problem in $H^1_0(\Omega)$ (without internal discontinuity interface). This allows proving the well-posedness of the problem in a direct way by the Lax–Milgram lemma [31].

Both formulations (2.8) and (2.4) are equivalent from the mathematical point of view. Note that \bar{u} and \hat{u} have different regularity. In fact, the jump of normal derivatives of \hat{u} is 0 across Γ whereas that of \bar{u} is not.

Obviously, their numerical approximation would yield different numerical solutions. We will discuss this issue in section 5.1, while we focus now on the construction of the lifting operators R_i and S_i .

2.1. The continuous lifting operators. Formulations (2.4) and (2.8) require the knowledge of liftings of the jump conditions. If such liftings are already provided with the definition of the problem, they can be used without modifications and this section might be skipped. However, since in general these liftings are not known, we propose an approach to construct them, which is based on the assumption that there exist two regular enough scalar functions \bar{g}_d and \bar{g}_n in Ω such that $g_d = \bar{g}_d|_\Gamma$ and $g_n = \bar{g}_n|_\Gamma$. We will discuss this point in section 2.2.

For the sake of simplicity, we start with the lifting for g_n which accounts for the jump in the normal derivative. From now on we assume that the level set function be such that $\phi \in C^1(\bar{\Omega})$. Consider the function

$$(2.9) \quad S g_n = H(\phi) \phi \bar{g}_n \quad \text{in } \Omega,$$

where $H(\phi)$ is the Heaviside function of the domain $\Omega \setminus \Omega_1$:

$$H(\phi)(x) = \begin{cases} 1 & \text{if } \phi(x) \geq 0, \\ 0 & \text{if } \phi(x) < 0, \end{cases}$$

whence

$$(2.10) \quad S g_n(x) = \begin{cases} \phi(x) \bar{g}_n(x) & \text{if } \phi(x) \geq 0, \\ 0 & \text{if } \phi(x) < 0. \end{cases}$$

Note that Sg_n is continuous across Γ (the latter being the 0 level set of ϕ), that is $\llbracket Sg_n \rrbracket_\Gamma = 0$. On the other hand

$$(2.11) \quad \left\llbracket \frac{\partial Sg_n}{\partial n} \right\rrbracket_\Gamma = \frac{\partial(\phi \bar{g}_n)}{\partial n}|_\Gamma = \frac{\partial \phi}{\partial n}|_\Gamma \bar{g}_n|_\Gamma + \frac{\partial \bar{g}_n}{\partial n}|_\Gamma \phi|_\Gamma = g_n,$$

thus Sg_n is a lifting of g_n which satisfies both conditions (2.7). We denote $(Sg_n)|_{\Omega_i}$ by $S_i g_n$.

Now we need a lifting Rg_d for the function g_d which is discontinuous across Γ although featuring a continuous normal derivative. We set

$$(2.12) \quad Rg_d = H(\phi)(\bar{g}_d - \phi \nabla \bar{g}_d \cdot \nabla \phi)$$

which can be expressed explicitly as

$$(2.13) \quad Rg_d(x) = \begin{cases} \bar{g}_d(x) - \phi(x) \nabla \bar{g}_d(x) \cdot \nabla \phi(x) & \text{if } \phi(x) \geq 0, \\ 0 & \text{if } \phi(x) < 0. \end{cases}$$

We can see that $\llbracket Rg_d \rrbracket_\Gamma = g_d$, whereas by a direct computation

$$(2.14) \quad \begin{aligned} \left\llbracket \frac{\partial Rg_d}{\partial n} \right\rrbracket_\Gamma &= \frac{\partial(\bar{g}_d - \phi \nabla \bar{g}_d \cdot \nabla \phi)}{\partial n}|_\Gamma \\ &= \frac{\partial \bar{g}_d}{\partial n}|_\Gamma - \frac{\partial(\nabla \bar{g}_d \cdot \nabla \phi)}{\partial n}|_\Gamma \phi|_\Gamma - \frac{\partial \phi}{\partial n}|_\Gamma (\nabla \bar{g}_d \cdot \nabla \phi)|_\Gamma \\ &= \frac{\partial \bar{g}_d}{\partial n}|_\Gamma - \frac{\partial(\nabla \bar{g}_d \cdot \nabla \phi)}{\partial n}|_\Gamma 0 - 1 (\nabla \bar{g}_d \cdot n)|_\Gamma \\ &= \frac{\partial \bar{g}_d}{\partial n}|_\Gamma - \frac{\partial \bar{g}_d}{\partial n}|_\Gamma = 0. \end{aligned}$$

Rg_d is therefore a lifting that satisfies conditions (2.6) (as before, we denote $R_i g_d = (Rg_d)|_{\Omega_i}$, $i = 1, 2$).

Remark 2.1. An alternative approach to that adopted before would consist in solving a suitable PDE in one of the subdomains: to control both the trace and the normal derivative at the same time, we can solve a fourth order biharmonic problem in Ω_2 . Precisely, the problem reads, find $L \in H^2(\Omega_2)$ such that

$$\begin{aligned} \Delta^2 L &= 0 & \text{in } \Omega_2, \\ L &= g_d & \text{on } \Gamma, \\ \frac{\partial L}{\partial n} &= g_n & \text{on } \Gamma, \\ L &= 0 & \text{on } \partial\Omega \setminus \Gamma, \\ \frac{\partial L}{\partial n} &= 0 & \text{on } \partial\Omega \setminus \Gamma. \end{aligned}$$

Instead of $Rg_d + Sg_n$ we would then consider $H(\phi)L$.

Example 2.1. We consider a one-dimensional (1D) example for the sake of clarity. The domain is $\Omega = (0, 1)$ and the interface Γ is composed of the two points $x_1 = \pi^{-1}$ and $x_2 = 1 - \pi^{-1}$. The level set function is defined then as $\phi(x) = \max(\pi^{-1} - x, x - 1 + \pi^{-1})$. We consider $\bar{g}_d(x) = \exp(2x)$ and $\bar{g}_n(x) = \sin(3x)$ so that $g_d(x_1) = \exp(2\pi^{-1})$,

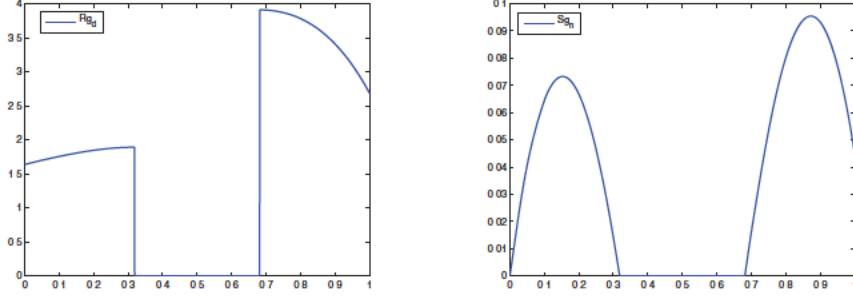


FIG. 2.1. Continuous liftings obtained for Example 2.1: Rg_d (left) and Sg_n (right).

$g_d(x_2) = \exp(2(1 - \pi^{-1}))$, $g_n(x_1) = \sin(3\pi^{-1})$, and $g_n(x_2) = \sin(3(1 - \pi^{-1}))$. The explicit formulas in this example are given by:

$$Rg_d = \begin{cases} \exp(2x) + 2\exp(2x)(\pi^{-1} - x) & \text{if } x < \pi^{-1}, \\ 0 & \text{if } \pi^{-1} < x < 1 - \pi^{-1}, \\ \exp(2x) - 2\exp(2x)(x - 1 + \pi^{-1}) & \text{if } 1 - \pi^{-1} < x, \end{cases}$$

$$Sg_n = \begin{cases} \sin(3x)(\pi^{-1} - x) & \text{if } x < \pi^{-1}, \\ 0 & \text{if } \pi^{-1} < x < 1 - \pi^{-1}, \\ \sin(3x)(x - 1 + \pi^{-1}) & \text{if } 1 - \pi^{-1} < x. \end{cases}$$

The continuous liftings Rg_d and Sg_n are shown in Figure 2.1.

2.2. Extending interface data. In section 2.1, we supposed that extensions of the interface data g_d and g_n (therein denoted by \bar{g}_d and \bar{g}_n , respectively) are already given and that they enjoy suitable regularity. This assumption is more or less strong depending on the way the data g_d and g_n are provided. If they are given as functions on the whole domain Ω or as finite element functions, this assumption is obviously fulfilled. This is the case, for example, when one considers surface tension in a two phase flow problem: indeed, surface tension, which creates a jump in the pressure (see [24]), can be expressed in terms of the level set function and its derivative and therefore can naturally be defined on the whole computational domain. In other cases, this is indeed a limitation and we provide here a method to overcome it. The function $\bar{g}_d \in H_0^1(\Omega)$ can be defined as $\bar{g}_d|_{\Omega_i} = \bar{g}_d^i$ in Ω_i ($i = 1, 2$), where $\bar{g}_d^i \in H_{\partial\Omega_i \setminus \Gamma}^2(\Omega_i)$ is the solution of the following biharmonic problem:

$$(2.15) \quad \begin{aligned} \Delta^2 \bar{g}_d^i &= 0 & \text{in } \Omega_i, \\ \bar{g}_d^i &= g_d & \text{on } \Gamma, \\ \bar{g}_d^i &= 0 & \text{on } \partial\Omega_i \setminus \Gamma, \\ \frac{\partial \bar{g}_d^i}{\partial n} &= 0 & \text{on } \partial\Omega_i. \end{aligned}$$

On the other hand, to construct the extension \bar{g}_n , we can consider the following harmonic problems:

$$(2.16) \quad \begin{aligned} -\Delta \bar{g}_n^i &= 0 & \text{in } \Omega_i, \\ \bar{g}_n^i &= g_n & \text{on } \Gamma, \\ \bar{g}_n^i &= 0 & \text{on } \partial\Omega_i \setminus \Gamma, \end{aligned}$$

and we set $\bar{g}_n|_{\Omega_i} = \bar{g}_n^i$, $i = 1, 2$.

We notice that the extension of g_d implies $R_i g_d \in H_{\partial\Omega_i \setminus \Gamma}^1(\Omega_i)$, while $\bar{g}_n^i \in H_{\partial\Omega_i \setminus \Gamma}^1(\Omega_i)$. This guarantees the well-posedness of the weak formulation (2.8). In section 3.1.1 we will discuss how to construct practical extensions of the interface data at the discrete level.

3. Finite element approximation. In this section, we will address the numerical approximation of the problems introduced thus far, together with the introduction of the approximate lifting operators.

We consider a uniform triangulation τ_h of Ω made of elements K (intervals if $N = 1$, triangles if $N = 2$, or tetrahedra if $N = 3$). The interface Γ may intersect the elements K arbitrarily. As a finite element space, we use the continuous \mathbb{P}_1 elements

$$(3.1) \quad V_h = \{v_h \in H_0^1(\Omega) \cap C^0(\bar{\Omega}) : v_h|_K \in \mathbb{P}_1 \ \forall K \in \tau_h\},$$

and we denote by $\{\Psi_j\}$ the basis functions of V_h .

The finite element approximation of (2.4) reads find $\bar{u}_h \in V_h$ such that

$$(3.2) \quad \int_{\Omega} \nabla \bar{u}_h \cdot \nabla v_h = \int_{\Gamma} g_n v_h + \sum_{i=1}^2 \int_{\Omega_i} f_i v_{hi} - \sum_{i=1}^2 \int_{\Omega_i} \nabla R_i^h g_d \cdot \nabla v_{hi} \quad \forall v_h \in V_h,$$

while that of (2.8) becomes find $\hat{u}_h \in V_h$ such that, $\forall v_h \in V_h$,

$$(3.3) \quad \int_{\Omega} \nabla \hat{u}_h \cdot \nabla v_h = \int_{\Gamma} g_n v_h + \sum_{i=1}^2 \int_{\Omega_i} f_i v_{hi} - \sum_{i=1}^2 \int_{\Omega_i} \nabla (R_i^h g_d + S_i^h g_n) \cdot \nabla v_{hi}.$$

3.1. Discrete lifting operators. We introduce now suitable finite element approximations of the continuous liftings Rg_d and Sg_n . At the discrete level, we would like to have liftings with minimal support around the interface. Ideally, only the cells crossed by the interface ought to be used in order to keep the computational cost of the finite element approximation as low as possible. This is why the knowledge of the extensions \bar{g}_d and \bar{g}_n will be required only in those neighboring cells.

Let $\pi_h^1 : H_0^1(\Omega) \rightarrow V_h$ be the classical finite element interpolation operator

$$(3.4) \quad \pi_h^1(v) = \sum_j v(x_j) \Psi_j,$$

i.e., $\pi_h^1(v)$ is the unique function in V_h which takes the same values of v at all finite element nodes x_j , Ψ_j being the characteristic basis function associated with x_j , that is, $\Psi_j \in V_h : \Psi_j(x_i) = \delta_{ij} \ \forall i, j$ (see [30]).

Remark that both liftings Rg_d and Sg_n that we have defined at the continuous level are the product of the Heaviside function by a suitable function. For the sake of simplicity, we assume now that such a function is continuous. (We will further discuss this point in Remark 3.1.) For all $t \in H_0^1(\Omega)$, define $T = H(\phi)t$ and then

$$(3.5) \quad \Pi_h^1(T)(x) = \begin{cases} \pi_h^1(t)(x) & \text{if } \phi(x) \geq 0, \\ 0 & \text{if } \phi(x) < 0. \end{cases}$$

Note that $\Pi_h^1(T) = \pi_h^1(t)H(\phi)$. We define then the discrete liftings $R_{glo}^h g_d = \Pi_h^1(Rg_d)$ and $S_{glo}^h g_n = \Pi_h^1(Sg_n)$. The index *glo* stands for *global* and it indicates that these functions are defined on the global domain Ω .

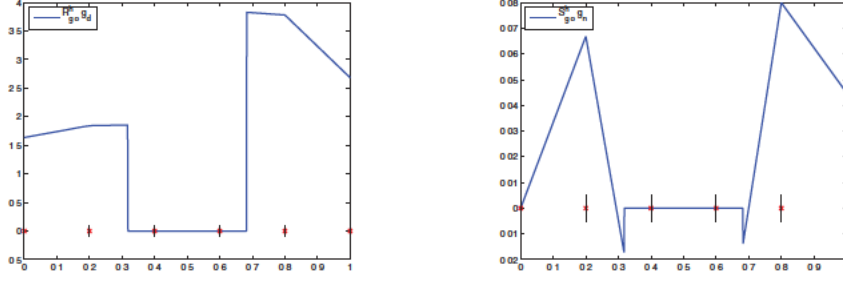


FIG. 3.1. Discrete global liftings $R_{glo}^h g_d$ (left) and $S_{glo}^h g_n$ (right). The crosses show the location of the degrees of freedom.



FIG. 3.2. Illustration of Ω_Γ in a two-dimensional (2D) case.

Example 3.1. With the same settings as Example 2.1, we perform the interpolation on a mesh with 5 intervals using \mathbb{P}_1 finite elements. The resulting liftings are shown in Figure 3.1.

To reduce the computational cost induced by the fact that our liftings have global support, we introduce a region Ω_Γ of width h around the interface Γ (see Figure 3.2) and we modify $R_{glo}^h g_d$ and $S_{glo}^h g_n$ so that the support of the modified functions is reduced to Ω_Γ . Notice that Ω_Γ corresponds to the strip of width h formed by those triangles that intersect the interface.

We use the abstract notation T_{glo}^h to identify either $R_{glo}^h g_d$ or $S_{glo}^h g_n$. We can express T_{glo}^h on each side of the interface using the finite element basis

$$(3.6) \quad T_{glo}^h(x) = \begin{cases} \sum_i \alpha_i \Psi_i(x) & \text{if } \phi(x) \geq 0, \\ 0 & \text{if } \phi(x) < 0. \end{cases}$$

Adding any arbitrary function from the finite element space on both sides of the interface does not change the jump of this lifting. Using the notation $\phi_i = \phi(x_i)$ and $\sum_{\phi_i \geq 0}$ to indicate $\sum_{\{i: \phi_i \geq 0\}}$ (analogously, we denote $\sum_{\phi_i < 0}$), we can define a new lifting T^h as

$$(3.7) \quad T^h(x) = \begin{cases} \sum_{\phi_i < 0} \alpha_i \Psi_i(x) & \text{if } \phi(x) \geq 0, \\ -\sum_{\phi_i \geq 0} \alpha_i \Psi_i(x) & \text{if } \phi(x) < 0. \end{cases}$$

By definition, the support of T^h is Ω_Γ and it is actually 0 on $\partial\Omega_\Gamma$ and it is then extended by zero outside Ω_Γ . Applying this procedure to $R_{glo}^h g_d$ and $S_{glo}^h g_n$ we obtain the liftings $R^h g_d$ and $S^h g_n$ that fulfill all our requirements.

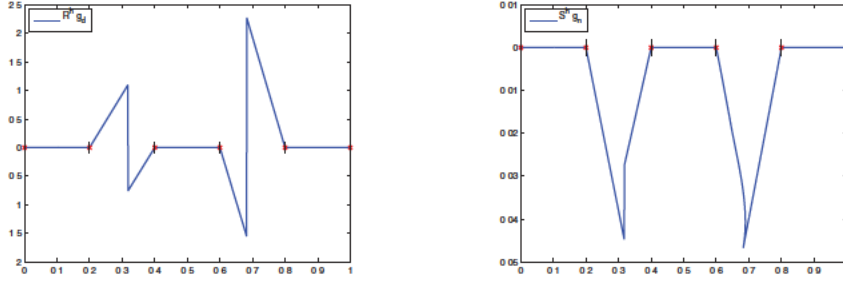


FIG. 3.3. Liftings after the support reduction: $R^h g_d$ (left) and $S^h g_n$ (right). The crosses show the location of the degrees of freedom. These liftings should be compared to those in Figure 3.1.

We can now give the explicit expression of the two liftings:

$$(3.8) \quad R^h g_d(x) = \begin{cases} \sum_{\phi_i < 0} (\bar{g}_d(x_i) - \nabla \bar{g}_d(x_i) \cdot \nabla \phi(x_i)) \phi(x_i) \Psi_i(x) & \text{if } \phi(x) \geq 0, \\ -\sum_{\phi_i \geq 0} (\bar{g}_d(x_i) - \nabla \bar{g}_d(x_i) \cdot \nabla \phi(x_i)) \phi(x_i) \Psi_i(x) & \text{if } \phi(x) < 0, \end{cases}$$

$$(3.9) \quad S^h g_n(x) = \begin{cases} \sum_{\phi_i < 0} (\bar{g}_n(x_i) \phi(x_i)) \Psi_i(x) & \text{if } \phi(x) \geq 0, \\ -\sum_{\phi_i \geq 0} (\bar{g}_n(x_i) \phi(x_i)) \Psi_i(x) & \text{if } \phi(x) < 0. \end{cases}$$

In case the interface coincides with the boundary of an element, the formulas (3.8) and (3.9) remain valid, in the sense that (2.6) and (2.7) are still approximated in a suitable way. In such a case, the correction will be taken into account only on those elements belonging to the subdomain characterized by negative values of the level set function.

Example 3.2. In Figure 3.3 we show the new liftings $R^h g_d$ and $S^h g_n$ of reduced support corresponding to the functions of the example 2.1 using \mathbb{P}_1 polynomials.

Remark 3.1. In case the functions multiplying the Heaviside function in the definition of the liftings Rg_d and Sg_n are not continuous but enjoy an $L^2(\Omega)$ - or $H^1(\Omega)$ -regularity, one can proceed similarly to define the discrete lifting operators $R^h g_d$ and $S^h g_n$. However, notice that instead of the interpolation operator (3.4) one could consider an elliptic (or Riesz) projector (see, e.g., [16, section 1.6.3]). The definition of the discrete liftings would change accordingly, but an explicit characterization like (3.8) and (3.9) would not be available.

3.1.1. Discrete extension of the interface data. In the previous section 3.1 we have assumed the discrete extensions \bar{g}_d and \bar{g}_n of the interface data to be known. In practice, this occurs in several cases, e.g., when treating pressure discontinuity in two phase flows, which is the main application that motivates this work (see [32, 13]). Other applications may be found, e.g., in solidification problems [9]. If \bar{g}_d and \bar{g}_n are already available for the problem at hand, one could directly follow the procedure presented in section 3.1. In this paper, we suppose to be in such a situation.

If this is not the case, then a preliminary step to construct such extensions should be considered. Extending interface data correctly deserves a whole study in itself, so we only mention here some possible solutions to this problem.

- A possible approach would be to compute finite element approximations of the continuous problems (2.15) and (2.16) using isogeometric methods. We refer to [12, 6, 5] for more details on this approach.
- Another possible strategy to compute the discrete extensions \bar{g}_d and \bar{g}_n would be to use the method proposed in [20], in the context of the ESIC method.

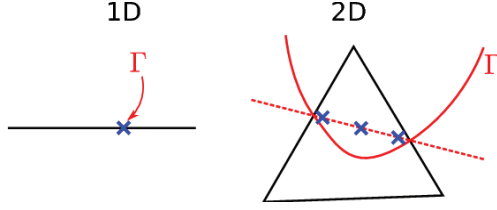


FIG. 3.4. Illustration of the methods used for the computation of the line integral in 1 and 2 dimensions.

In that case, the extension \bar{g}_* on Ω_Γ of a generic function g_* defined only on Γ is defined as follows:

$$(3.10) \quad \bar{g}_*(x) = g_*(x_\Gamma) \quad \forall x \in \Omega_\Gamma,$$

where x_Γ is the point of Γ that minimizes the distance to x . As a consequence, \bar{g}_* is constant along any normal direction issuing from Γ . In case the computational mesh is structured, a possible way of building the extension \bar{g}_* is provided in [10].

- Finally, another possibility is to solve a PDE evolving with a pseudotime, as proposed in [28]. The drawbacks of this approach are that the necessary regularities of the extension cannot be ensured and that an accurate numerical approximation of the solution might be quite difficult to obtain near the interface.

Remark that, thanks to the reduction of the support that we have proposed in section 3.1 (see (3.7)), the extensions \bar{g}_d and \bar{g}_n of the interface data g_d and g_n have to be known only on Ω_Γ . This potentially reduces the cost of any extension procedure.

3.2. The SESIC method. The SESIC method that we propose is obtained by using the discrete lifting operators (3.8) and (3.9) in the context of the weak formulation (3.3). The only ingredient that remains to be detailed is the numerical integration formula that will be used to compute the new terms in the weak formulation (3.3). More precisely, to compute the right-hand side, we have to perform one integral on the interface Γ and two integrals over Ω of possibly discontinuous functions (indeed, both $\nabla R^h g_d$ and $\nabla S^h g_n$ might be discontinuous across Γ at the discrete level). We propose two different methods for the integration.

Two-side integration. The first method consists in building quadrature rules that take into account the interface. A possible way to integrate singular functions of type

$$\int_\Gamma f = \int_\Omega \delta_\Gamma f$$

is to reconstruct the interface Γ explicitly and then to use on it a $(N-1)$ -dimensional quadrature rule. If $N=1$, the interface reduces to a point and the integration requires only to evaluate f at a given point. If $N=2$, the elements are triangles and then the interface in a single triangle is a segment in the case of a piecewise linear approximation. A schematic representation is shown in Figure 3.4. To apply a suitable integration rule on this segment, we need to compute the intersections of Γ with the edges of the triangle.

On the other hand, to integrate discontinuous functions like

$$\int_\Omega f_1 + H(\phi)f_2,$$

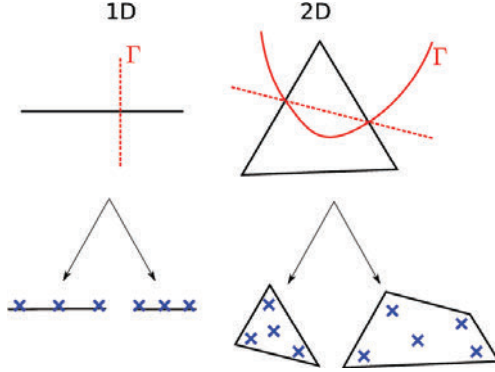


FIG. 3.5. Illustration of the methods used for the computation of the discontinuous functions in 1 and 2 dimensions.

we define a quadrature rule for an element crossed by Γ considering a quadrature rule on the polygons on each side of the interface. More precisely, if $N = 1$, we compute the location of the interface then we combine a quadrature rule for segments on each side of the interface. When $N = 2$, the triangles crossed by Γ are split into a triangle and a quadrilateral. To integrate discontinuous functions, we combine then a quadrature rule for triangles and a quadrature rule for quadrilaterals (see Figure 3.5).

Similar methods are available for three-dimensional (3D) simulations, however they lead to complicated schemes where many different cases have to be distinguished depending on the way the interface cuts the tetrahedra. Thus we propose an alternative method that, besides being much simpler, is more suitable for higher space dimensions or for higher polynomial degrees.

Integration of regularized functions. The idea, that can be adopted in any space dimension, is to approximate singular or discontinuous integrands by smooth functions. For example, we make the following approximation:

$$\int_{\Gamma} g_n v = \int_{\Omega} \bar{g}_n v \delta_{\Gamma} \cong \int_{\Omega} \bar{g}_n v \delta_w,$$

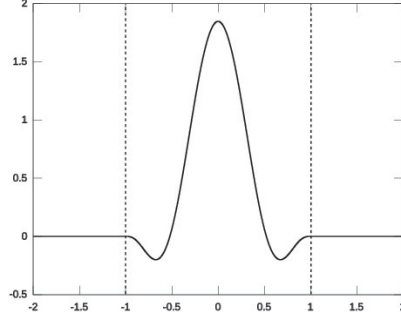
where δ_w is an approximation of δ_{Γ} whose support is limited to a band of width w around Γ . This method is quite widely used, even if, often, there is no real control on the error produced. We refer to [34] for the error analysis of the regularization step, in which two errors are highlighted:

- the error (called “analytical error” in [34]) produced by the introduction of the regularizing function

$$\left| \int_{\Gamma} g_n v - \int_{\Omega} \bar{g}_n v \delta_w \right|;$$

- the quadrature error coming from the inexact integration of the regularized integrand.

The usual procedure is to take w proportional to h . However, the approach that we adopt in this paper is to use a width w that is proportional to \sqrt{h} (a similar width has been suggested in [24] in another context). Our choice is motivated by the following considerations. First of all, our function δ_w must have the form $\delta_w(d) = \frac{1}{w} \hat{\delta}(d/w)$ (prolongated by 0 outside the band of width w), where $\hat{\delta}$ is a function that

FIG. 3.6. Plot of the function $\hat{\delta}(d)$.

does not depend on w , the factor $\frac{1}{w}$ making the weight of δ_w constant with respect to w , and d is the distance to the interface, that is $d(x) = \phi(x)$.

- The analytical error is then proportional to w^β , where β can be computed using the properties (vanishing moments) of $\hat{\delta}$ [34].
- In each triangle we have to integrate the finite element function $\bar{g}_n^h v_h \delta_w$ whose regularity depends on the regularity of δ_w . We can distinguish two cases: if a triangle K intersects one of the curves $\phi = \pm w$, then $\bar{g}_n^h v_h \delta_w$ inherits the same regularity of δ_w , i.e., $\bar{g}_n^h v_h \delta_w \in C^m(K)$, provided $\delta_w \in C^m(K)$; otherwise, $\bar{g}_n^h v_h \delta_w \in C^\infty(K)$. Given a finite element mesh of size h , if we adopt a quadrature rule with a degree of exactness $m-1$ (see [29]), the quadrature error is proportional to $h^{m+2} \|(\bar{g}_n^h v_h \delta_w)^{(m+1)}\|_{L^\infty(\Omega)}$ [34]. The derivative of order $(m+1)$ of the integrand function can be computed provided δ_m has been chosen regular enough. Moreover, since $\delta_w^{(m+1)}$ scales like $w^{-(m+2)}$, the quadrature error will behave like $(h/w)^{m+2}$.

Based on these arguments, we conclude that, by choosing $w = h$, we cannot ensure that the quadrature error will vanish when $h \rightarrow 0$ since the number of quadrature points in the band of width w would be constant while the function δ_w gets steeper to conserve the mass. Our choice of $w = c\sqrt{h}$ leads to control of the analytical error by $h^{\beta/2}$ and the quadrature error by $h^{(m+2)/2}$. We can then fully control the decay rate of the overall error by choosing the appropriate $\hat{\delta}$ function.

If we look for second order accuracy, building $\hat{\delta}$ with 3 vanishing moments (then $\beta = 4$, see [34]) and 2 continuous derivatives would be sufficient. By looking for the polynomial function with the smallest degree featuring these properties, we end up with

$$(3.11) \quad \hat{\delta}(d) = \frac{315}{512} (3 - 20d^2 + 42d^4 - 36d^6 + 11d^8).$$

A representation of this function is given in Figure 3.6.

The same approach can be applied for the Heaviside function to integrate a discontinuous integrand across the interface. We used for our tests the regularized Heaviside function

$$\hat{H}(d) = \int_{-1}^d \hat{\delta}(\xi) \, d\xi.$$

To illustrate our choice for the integration, we consider 3 examples in two dimensions, where the domain is the square $\Omega = (-1, 1)^2$ and the triangulation is made by triangles whose typical size is denoted by h . All the integrations are performed using

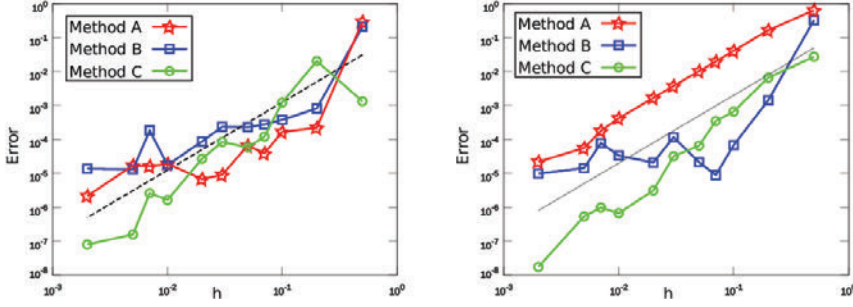


FIG. 3.7. Results of Examples 3.3 (left) and 3.4 (right).

a Dunavant quadrature rule of degree 4 in all the triangles of the mesh [15]. For each example, we will use three different methods to evaluate a line integral on the 0 level set of the function ϕ :

- **Method A.** The bandwidth is set to $w = 2h$ and we take $\hat{\delta}(d) = (1 + \cos(\pi d))/2$.
- **Method B.** The bandwidth is set to $w = 2h$ and $\hat{\delta}$ is defined as in (3.11).
- **Method C.** The bandwidth is set to $w = \sqrt{h}/2$ and $\hat{\delta}$ is again as in (3.11).

Example 3.3. The first example consists in simply calculating the length of a circle. The level set function is defined as $\phi(x, y) = \sqrt{x^2 + y^2} - 0.5$. In this example, the error is due only to the quadrature error. Figure 3.7 (left) shows the magnitude of the error as a function of the mesh size h . We can remark that with methods A and B the convergence quickly slows down and does not seem to converge in the end, while method C yields a convergence rate of order 2.

Example 3.4. The second example consists in integrating on a quarter of a circle the function $f(x, y) = (y + 1)\exp(x + 1)$. The level set function is $\phi(x, y) = \sqrt{(x + 1)^2 + (y + 1)^2} - 1.5$. Figure 3.7 (right) shows the results obtained. We can see that the methods A and B give rise to a good convergence for coarse meshes, but the convergence slows down at a level comparable to the one observed in Example 3.3. Method C is more robust as it yields again a convergence rate close to 2 for the whole range of meshes tested.

Example 3.5. The third example consists again in computing the length of a curve. However, the curve that we consider has just a C^0 regularity (see Figure 3.8 (left)). (Remark that in this case $\partial\phi/\partial n$ is well defined in all the finite element nodes, because the mesh is not conforming with the kinks of the interface.) As shown in Figure 3.8 (right), the convergence rates for the methods A, B, and C are lower than in the previous examples. We can also remark that method C yields a slower convergence in this example.

Example 3.6. The last example compares the quality of the approximation obtained with the interpolant $\pi_h^1\phi$ of the level set function ϕ instead of using the continuous level set function ϕ itself as we have done so far. Considering the level set function $\phi(x, y, z) = (x^2 + y^2 + z^2)^{1/2} - 0.5$, we approximate the surface of a 3D sphere and of half of a sphere by integrating $g_n(x, y, z) = 1$ and $g_n(x, y, z) = \mathbb{I}_{x < y}$, respectively (\mathbb{I} being the indicator function). In Figure 3.9, we can observe that, even with the interpolated level set function, method A yields second order convergence for both tests.

From these examples, we can see that in the case the level set function has a lower regularity than $C^1(\bar{\Omega})$, as we assumed in this paper, method C might underperform

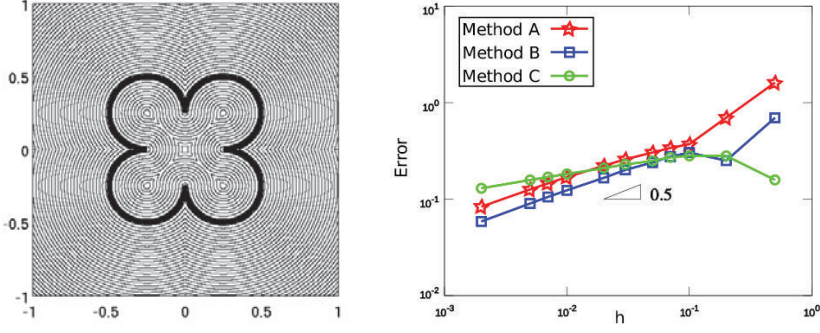


FIG. 3.8. Representation of the isocurves of the level set function (left, Γ in bold) and magnitude of the error (right) on the computation of the length of the curve.

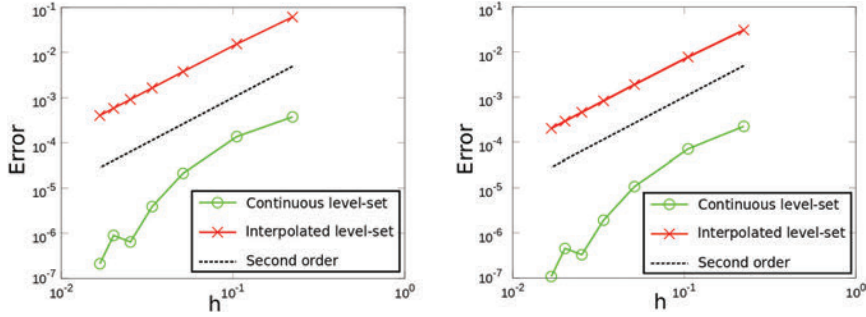


FIG. 3.9. Magnitude of the error on the computation of the surface of the sphere (left) and half of the sphere (right) with method A.

with respect to methods *A* and *B*. Indeed, when the regularity is low, the integration error might be controlled by a term of the form w^α , where $\alpha > 0$ is relatively small (around 0.5 in Example 3.5). In such a case, choosing w proportional to \sqrt{h} might give worse results than if it is proportional to h . Similar remarks hold for 3D cases.

3.3. On the choice of the discrete lifting operator. In the previous section 3.1, we proposed a particular construction of the lifting operators that is interesting as it does not require us to reconstruct the interface. We shall demonstrate here that the solution u_h obtained using the SESIC method is actually independent of this construction to a certain extent.

Suppose that we consider a suitable discrete lifting $\tilde{T}^h = \tilde{T}^h(g_d, g_n)$ such that $R^h g_d + S^h g_n - \tilde{T}^h \in V_h$. As a consequence

- \tilde{T}^h is linear in each subdomain Ω_i , $i = 1, 2$;
- the interface conditions are satisfied in the same way as with our construction in the sense that they satisfy the same jumps across the interface at the discrete level.

Using the discrete formulation of the SESIC method (3.3), we denote by $\hat{w}_h \in V_h$ the solution obtained considering the lifting \tilde{T}^h , i.e., $\forall v_h \in V_h$,

$$(3.12) \quad \int_{\Omega} \nabla \hat{w}_h \cdot \nabla v_h = \int_{\Gamma} g_n v_h + \sum_{i=1}^2 \int_{\Omega_i} f_i v_{hi} - \sum_{i=1}^2 \int_{\Omega_i} \nabla \tilde{T}^h \cdot \nabla v_{hi}.$$

We now prove that $w_h = \hat{w}_h + \tilde{T}^h$ coincides with u_h . Subtracting (3.12) from (3.3), we obtain, $\forall v_h \in V_h$,

$$(3.13) \quad \int_{\Omega} \nabla(u_h - w_h) \cdot \nabla v_h = 0.$$

We have $u_h - w_h = (\hat{u}_h + Rg_d + Sg_n) - (\hat{w}_h + \tilde{T}^h) \in V_h$ as by definition $\hat{u}_h \in V_h$, $\hat{w}_h \in V_h$, and $R^h g_d + S^h g_n - \tilde{T}^h \in V_h$ by assumption. Since the Galerkin approximation of the Laplace problem is well-posed on V_h , from (3.13) it follows that $u_h - w_h = 0$.

In view of this result, we can now comment on the case of 1D problems. As already remarked, using the extension of the interface data proposed in section 2.2, \bar{g}_d and \bar{g}_n are constant in the vicinity of the interface (that might be composed of several points). Thanks to the definitions (2.9) and (2.12) and to the fact that the level set function ϕ is piecewise linear in one dimension, Rg_d and Sg_n are piecewise linear in the neighborhood of the interface. Therefore, the interpolation step (3.5) leaves Rg_d and Sg_n unchanged and the jump conditions (2.6) and (2.7) are satisfied exactly. This implies that in one dimension, any construction of the liftings using piecewise linear functions and satisfying exactly the interface conditions yields the same solution as the SESIC method.

Based on this argument we can conclude, that in the 1D case, the SESIC method is a special case of the immersed finite element method (IFEM) formulated in [19]. More precisely

1. in SESIC, the right-hand side contribution coming from the lifting is not computed using a special basis function, but through an explicit formula;
2. in the IFEM, different physical parameters k_i can appear in the original equation (1.1), i.e., $-\nabla \cdot (k_i \nabla u_i) = f_i$ in Ω_i , while they must be the same for the SESIC method to work;
3. for multidimensional problems, the two methods do not coincide.

3.4. The ESIC method. As stated before, the SESIC method is inspired by the ESIC method first proposed in [20]. For the sake of comparison, let us recall the principle of the ESIC method and emphasize the differences with the new SESIC method.

The two methods are built on different weak formulations of the given problem (1.1)–(1.3). SESIC stems from the weak form (3.3) whereas in order to get rid of the line integral in (3.3), Huh and Sethian perform a counter integration by parts yielding

$$(3.14) \quad \int_{\Gamma} g_n v_h - \sum_{i=1}^2 \int_{\Omega_i} \nabla S_i g_n \cdot \nabla v_{hi} = \sum_{i=1}^2 \sum_K \left(\int_{K \cap \Omega_i} \Delta S_i g_n v_{hi} - \int_{\partial K} \frac{\partial S_i g_n}{\partial n} v_{hi} \right).$$

However, this formulation introduces a new error source, as it makes use of the equality $g_n = \llbracket \frac{\partial S g_n}{\partial n} \rrbracket$, which might be inexact at the discrete level. This is in fact documented by the numerical tests that we will present in section 5.

Like the SESIC method, the ESIC method requires data to be extended from the interface to build the lifting operators. On this subject, the ESIC is slightly more restrictive than the SESIC method, since it requires an extension that is constant in the direction normal to the interface.

A last major difference between the SESIC and the ESIC methods is the construction of the lifting operator for the jump in the normal derivative. Indeed, the

discrete lifting operator proposed for the ESIC reads

$$(3.15) \quad \tilde{S}^h g_n = \begin{cases} \sum_{\phi_i < 0} (\bar{g}_n(x_i)) \Psi_i(x) \phi(x) & \text{if } \phi(x) \geq 0, \\ -\sum_{\phi_i \geq 0} (\bar{g}_n(x_i)) \Psi_i(x) \phi(x) & \text{if } \phi(x) < 0 \end{cases}$$

(with the same notations as in (3.9)). This differs from the definition (3.9) by the fact that in (3.15) not only the values of the level set function at the nodes are used, but the whole function is used to define the discrete lifting. On one side, this ensures that the lifting $\tilde{S}^h g_n$ is continuous across the interface at the discrete level, but the potential drawback of this construction is that the multiplication by ϕ increases the polynomial order of the lifting function, thus requiring a polynomial refinement in the neighborhood of the interface.

3.5. Higher order approximations. Our construction concerns only linear finite elements. However, it can be generalized to higher order polynomial approximation. Indeed, only the finite element space (3.1) and the interpolation operator (3.5) have to be adapted.

The reason for treating only the linear case resides in the regularity of \hat{u} . Indeed, thanks to the construction of the liftings, we know that $\hat{u} \in H^2(\Omega)$, but it is not possible to ensure more regularity, e.g., $H^3(\Omega)$, as the liftings introduced so far do not provide control on second order derivatives across the interface. The use of a higher order polynomial for the approximation would not be necessarily rewarded by an higher convergence rate, in particular, close to the interface.

In order to achieve a higher convergence rate, one would need to build liftings that correct the second (and possibly higher) derivatives. This can be done by considering functions of the type [21]

$$(3.16) \quad L_k g = H(\phi) \phi^k g$$

which have $k-1$ continuous derivatives across Γ and yield a jump in the k th derivative. The solution would then be further decomposed, following (2.1) and (2.5):

$$(3.17) \quad u_i = \bar{u}_i + R_i g_d + S_i g_n + \sum_{j=2}^l L_i^j g$$

and the weak formulation would be changed accordingly.

If one considers the level set function as discretized, a high order approximation for the level set function must also be considered. This can greatly increase the intricacy of the construction of the two-sided integration.

4. Error analysis. For the sake of analysis, besides the regularity assumptions made in the previous sections, we also suppose H^2 regularity, i.e., we suppose that in (2.8) we have $\hat{u} \in H^2(\Omega) \cap H_0^1(\Omega)$ and that the restrictions of the continuous liftings Rg_d and Sg_n to Ω_i belong to $H^2(\Omega_i)$, $i = 1, 2$. Throughout this section we will denote by C a suitable constant which may change in the different inequalities, but that will always be independent of h .

In this section, we carry out the error analysis of the SESIC method, using the weak formulation (3.3). To perform the analysis, we will use exclusively the liftings $R_{glo}^h g_d$ and $S_{glo}^h g_n$, as they have continuous counterparts to which they can be compared. However, in practice, one would rather use the liftings $R^h g_d$ and $S^h g_n$. The following argument shows that our analysis also stands for this latter couple of liftings.

Let us denote by \hat{u}_h the solution of the problem (3.3) using $R^h g_d$ and $S^h g_n$, $u_h = \hat{u}_h + R^h g_d + S^h g_n$, \hat{u}_h^{glo} the solution of problem (3.3) using $R_{glo}^h g_d$ and $S_{glo}^h g_n$

and $u_h^{glo} = \hat{u}_h^{glo} + R_{glo}^h g_d + S_{glo}^h g_n$. First of all, we remark that $u_h - u_h^{glo} \in V_h$, as $\hat{u}_h, \hat{u}_h^{glo} \in V_h$ by definition and $R^h g_d - R_{glo}^h g_d \in V_h$, $S^h g_n - S_{glo}^h g_n \in V_h$ because of the way the support reduction has been performed in section 3.1. Since both \hat{u}_h and \hat{u}_h^{glo} satisfy (3.3), we have

$$(4.1) \quad \int_{\Omega} \nabla(u_h - u_h^{glo}) \cdot \nabla v_h = 0 \quad \forall v_h \in V_h.$$

Since the Galerkin approximation of the Laplace problem is well-posed on V_h , $u_h - u_h^{glo}$ is the unique solution. As 0 is also a solution of that problem, these two solutions must coincide: $u_h - u_h^{glo} = 0$. Therefore, the two lifting sets yield the same final solution and so the convergence analysis carried out for u_h^{glo} also applies to u_h .

In order to obtain a priori error estimates, we split the error into three parts:

$$(4.2) \quad u - u_h^{glo} = (\hat{u} + Rg_d + Sg_n) - (\hat{u}_h^{glo} + R_{glo}^h g_d + S_{glo}^h g_n)$$

whence

$$(4.3) \quad \|u - u_h^{glo}\| \leq \|\hat{u} - \hat{u}_h^{glo}\| + \|Rg_d - R_{glo}^h g_d\| + \|Sg_n - S_{glo}^h g_n\|,$$

where $\|\cdot\|$ represents a suitable norm. Using the construction of the lifting operators and the classical interpolation error estimates for the operator π_h^1 (see, e.g., [16] for either the interpolation operator (3.4) or the projection operator introduced in Remark 3.1), the last two terms may be bounded as follows: if T (respectively, T_{glo}^h) denotes either Rg_d or Sg_n (respectively, R_{glo}^h or S_{glo}^h), we have

$$\sum_{i=1}^2 \|T - T_{glo}^h\|_{H^1(\Omega_i)} = \|T - T_{glo}^h\|_{H^1(\Omega_2)},$$

thanks to (3.4) and to the fact that $T = H(\phi)t$. Therefore, we also have

$$\|T - T_{glo}^h\|_{H^1(\Omega_2)} = \|t - \pi_h^1(t)\|_{H^1(\Omega_2)} \leq \|t - \pi_h^1(t)\|_{H^1(\Omega)}.$$

Using standard interpolation results [16], we obtain

$$(4.4) \quad \sum_{i=1}^2 \|T - T_{glo}^h\|_{H^1(\Omega_i)} \leq Ch \sum_{i=1}^2 \|t\|_{H^2(\Omega_i)}.$$

It remains to estimate the first term on the left-hand side of (4.3). The analysis that we will carry out in this section does not rely on a specific construction of the lifting operators and could be applied to other constructions.

4.1. Convergence in the broken H^1 norm. To quantify the convergence in the energy norm, we use the first Strang lemma [30] (in our case, the bilinear forms of the continuous and discrete weak formulations coincide):

$$(4.5) \quad \begin{aligned} \|\hat{u} - \hat{u}_h^{glo}\|_{H^1(\Omega)} &\leq \left(1 + \frac{\gamma}{\alpha}\right) \inf_{w_h \in V_h} \|\hat{u} - w_h\|_{H^1(\Omega)} \\ &\quad + \frac{1}{\alpha} \sup_{v_h \in V_h} \frac{1}{\|v_h\|_{H^1(\Omega)}} \left| \sum_{i=1}^2 \int_{\Omega_i} \nabla(Rg_d + Sg_n) \cdot \nabla v_h \right. \\ &\quad \left. - \sum_{i=1}^2 \int_{\Omega_i} \nabla(R_{glo}^h g_d + S_{glo}^h g_n) \cdot \nabla v_h \right|, \end{aligned}$$

where γ is the continuity constant of the bilinear form $a(u, v) = \int_{\Omega} \nabla u \cdot \nabla v$ and α its coercivity constant. The first term on the right-hand side can be bounded thanks to the properties of the finite element space used, i.e., linear Lagrangian functions:

$$(4.6) \quad \inf_{w_h \in V_h} \|\hat{u} - w_h\|_{H^1(\Omega)} \leq Ch|\hat{u}|_{H^2(\Omega)}.$$

For the second term, using the Cauchy–Schwarz inequality, we obtain

$$\begin{aligned} & \left| \sum_{i=1}^2 \int_{\Omega_i} \nabla(Rg_d + Sg_n) \cdot \nabla v_h - \int_{\Omega_i} \nabla(R_{glo}^h g_d + S_{glo}^h g_n) \cdot \nabla v_h \right| \\ &= \left| \sum_{i=1}^2 \int_{\Omega_i} \nabla(Rg_d - R_{glo}^h g_d + Sg_n - S_{glo}^h g_n) \cdot \nabla v_h \right| \\ &\leq \left(\sum_{i=1}^2 |(Rg_d - R_{glo}^h g_d) + (Sg_n - S_{glo}^h g_n)|_{H^1(\Omega_i)} \right) \|v_h\|_{H^1(\Omega)}. \end{aligned}$$

Therefore, thanks to the triangular inequality and the error estimate (4.4), we have

$$\begin{aligned} & \sup_{v_h \in V_h} \frac{1}{\|v_h\|_{H^1(\Omega)}} \left| \sum_{i=1}^2 \left(\int_{\Omega_i} \nabla(Rg_d + Sg_n) \cdot \nabla v_h - \int_{\Omega_i} \nabla(R_{glo}^h g_d + S_{glo}^h g_n) \cdot \nabla v_h \right) \right| \\ &\leq \sum_{i=1}^2 (|(Rg_d - R_{glo}^h g_d) + (Sg_n - S_{glo}^h g_n)|_{H^1(\Omega_i)}) \\ &\leq Ch(|\bar{R}g_d|_{H^2(\Omega)} + |\bar{S}g_n|_{H^2(\Omega)}), \end{aligned}$$

where we have denoted $\bar{R}g_d = \bar{g}_d - \phi \nabla \bar{g}_d \cdot \nabla \phi$ and $\bar{S}g_n = \phi \bar{g}_n$. To sum up, we have

$$(4.7) \quad \|\hat{u} - \hat{u}_h^{glo}\|_{H^1(\Omega)} \leq Ch(|\hat{u}|_{H^2(\Omega)} + |\bar{R}g_d|_{H^2(\Omega)} + |\bar{S}g_n|_{H^2(\Omega)})$$

so that \hat{u}_h^{glo} converges towards \hat{u} with order $O(h)$ in H^1 norm, and thanks to (4.3), we have also optimal convergence of u_h^{glo} towards u :

$$(4.8) \quad \sum_{i=1}^2 \|u - u_h^{glo}\|_{H^1(\Omega_i)} \leq Ch(|\hat{u}|_{H^2(\Omega)} + |\bar{R}g_d|_{H^2(\Omega)} + |\bar{S}g_n|_{H^2(\Omega)}).$$

5. Numerical results. In this section, we present numerical results obtained using the methodologies described in the previous sections for different geometric dimensions.

5.1. 1D test case. First of all, we consider a 1D Poisson problem, as this allows us to make complete error measurements and visualizations. We consider the unit interval $\Omega = (0, 1)$ with an interface located in $\Gamma = \{\pi^{-1}\}$ so that the uniform meshes that we will use will not conform with the interface. The level set function is defined as $\phi(x) = \pi^{-1} - x$. The Poisson problem consists in finding $u : \Omega \rightarrow \mathbb{R}$ such that $-u''(x) = -e^x$ for $0 < x < 1$, with boundary conditions $u(0) = 1$, $u(1) = e + 2$, and with the jump conditions

$$(5.1) \quad \llbracket u \rrbracket_{\Gamma} = -2\pi^{-1}, \quad \left\llbracket \frac{\partial u}{\partial n} \right\rrbracket_{\Gamma} = 2.$$

TABLE 5.1
Convergence rates for the different jumps.

Elements	Convergence rates for errors on:			
	$\llbracket Rg_d \rrbracket$	$\llbracket \frac{\partial Rg_d}{\partial n} \rrbracket$	$\llbracket Sg_n \rrbracket$	$\llbracket \frac{\partial Sg_n}{\partial n} \rrbracket$
\mathbb{P}_1	3	2	2	1
\mathbb{P}_2	3	2	3	2

The exact solution reads

$$u = \begin{cases} e^x & \text{if } x \leq \pi^{-1}, \\ e^x + 2x & \text{if } x > \pi^{-1}. \end{cases}$$

As both jump conditions are nonhomogeneous, we need to extend them in the whole domain Ω . To this aim, we define two possible sets of extensions to highlight the role of the choice of the extensions for the convergence of the method. The first set is made of arbitrary functions

$$(5.2) \quad \bar{g}_d(x) = -(2\pi^{-1} + \sin(x - \pi^{-1})), \quad \bar{g}_n(x) = 1 + e^{(x - \pi^{-1})}$$

while the second set, called simplified extensions, is made of constant functions

$$(5.3) \quad \bar{g}_d(x) = -2\pi^{-1}, \quad \bar{g}_n(x) = 2.$$

For the simplified extensions (5.3), thanks to the definitions (2.10) and (2.13), it is easy to see that the interpolation does not introduce any error while if we take the extensions in (5.2), the interpolation will produce some error on the jumps and the conditions (5.1) will not be satisfied exactly. In the latter case, we have measured the error due to the liftings on the jump conditions (5.1) for \mathbb{P}_1 and \mathbb{P}_2 finite elements. Table 5.1 shows the order of convergence of these errors for $h \rightarrow 0$.

We can see that the orders are optimal for all the quantities and that we have a superconvergence for Rg_d with \mathbb{P}_1 elements. This is because of the special circumstance that we are interpolating a function with zero derivative.

We compute now the approximation of the solution of this test case using different methods. To measure the associated error, we use

- the broken H^1 norm in the domain Ω , $\|\cdot\|_{H^1(\Omega_1)} + \|\cdot\|_{H^1(\Omega_2)}$,
- the broken L^2 norm in the domain Ω , $\|\cdot\|_{L^2(\Omega_1)} + \|\cdot\|_{L^2(\Omega_2)}$,
- the L^∞ norm in the entire domain Ω , that is represented by the maximum error computed in $10n - 1$ equidistributed points in Ω (n being the number of subintervals used).

5.1.1. Solution using the SESIC method. We apply now the SESIC method to solve the 1D problem, using the adapted integration from section 3.2. With this method, we typically get a quite smooth error pattern on the whole domain, as shown in Figure 5.1 (top left), that provides evidence that all the components of the error are balanced.

However, as shown in Table 5.2, we do not get optimal orders for the maximum error with \mathbb{P}_2 elements, while the errors in Ω and the errors for the \mathbb{P}_1 elements exhibit optimal convergence rates in the different norms used.

Optimal orders also in the L^∞ norm for \mathbb{P}_2 approximations can be recovered if we use the simplified extensions (5.3). The suboptimality remarked for \mathbb{P}_2 might be due to a lack of regularity of \hat{u} : we can ensure that $\hat{u} \in H^2(\Omega)$ but we would need to build liftings taking into account also the second derivatives to provide more regularity (see section 3.5).

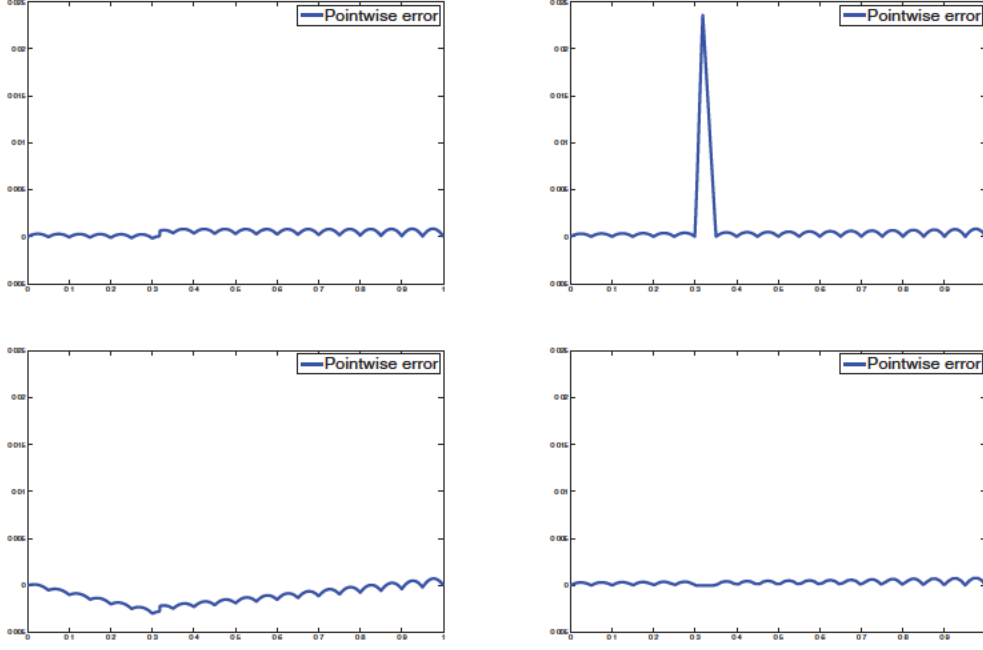


FIG. 5.1. Pointwise error in the solution for the 1D test using \mathbb{P}_1 elements and a grid of $n = 20$ intervals with the SESIC method (top left), with formulation (3.2) (top right), with the modification (3.14) (bottom left), and with the ESIC method (bottom right).

TABLE 5.2
Convergence rates for 1D problem with SESIC method.

Norm	\mathbb{P}_1 elements	\mathbb{P}_2 elements
$L^2(\Omega)$	3	
$H^1(\Omega)$	1	2
$L^\infty(\Omega)$	2	2

5.1.2. Solution based on the formulation (3.2). Using this test case, we can also provide a justification for the use of the lifting for the jump in the normal derivative. If we do not take into account the lifting Sg_n , we have to use the weak formulation (3.2). We keep the same definition for Rg_d (with the extension given in (5.2)) and we adopt the integration defined in section 3.2.

This formulation produces results that are different from the SESIC method mainly near the interface: Figure 5.1 (top right) shows a large error peak in the element crossed by the interface. The error located in that element is far larger than the interpolation error visible in the other elements.

This additional error comes from the fact that the underlying finite element space cannot reproduce jumps inside the elements. It is then impossible to reduce this error without providing the finite element space with the ability to capture jumps. In the SESIC method, this is the role of the lifting, that carries the jumps but does not belong to the finite element space. We can also see this behavior in Table 5.3 that shows the convergence orders for the method without Sg_n . The convergence of the error in all the considered norms is decreased because of the poor resolution of the solution near the interface.

NUMERICAL APPROXIMATION OF INTERFACE PROBLEMS

TABLE 5.3
Convergence rates for 1D problem using formulation (3.2).

Norm	\mathbb{P}_1 elements	\mathbb{P}_2 elements
$L^2(\Omega)$	1.5	1.5
$H^1(\Omega)$	0.5	0.5
$L^\infty(\Omega)$	1	1

TABLE 5.4
Convergence rates for 1D problem using the integration by parts (3.14).

Norm	\mathbb{P}_1 elements	\mathbb{P}_2 elements
$L^2(\Omega)$	1	2
$H^1(\Omega)$	1	2
$L^\infty(\Omega)$	1	2

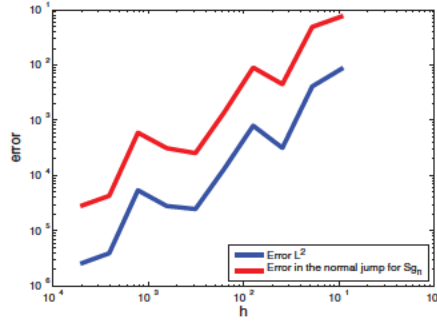


FIG. 5.2. Comparison between the pointwise error $|\llbracket \frac{\partial Sg_n}{\partial n} \rrbracket - g_n|$ on the normal jump of Sg_n and the $L^2(\Omega)$ error.

The role of the lifting Sg_n for the normal derivative is then clear: it helps to reduce the magnitude of the error in the neighborhood of the interface.

5.1.3. Solution based on (3.14). For the sake of comparing our method with the ESIC method described in section 3.4, we have also tested the weak formulation modified with (3.14) and with the liftings described in section 3.1. This allows us to bring to light the consequences of using (3.14) (we will use the two-side integration (see section 3.2) to keep the error coming exclusively from the method and not from the integration scheme). Table 5.4 shows the convergence rates for this test case; we can clearly see that the convergences are slower than with the weak formulation (3.3).

Figure 5.1 (bottom left) shows the typical pattern that we get using the modified weak formulation. The solution looks as if the force applied on the interface (by the term $\int_\Gamma g_n v_h$ in (3.2) and (3.3)) was badly estimated, leading to the trend of the error to be greater near the interface, while producing no peak there.

The origin of the error is also emphasized in Figure 5.2, that shows that there is a big correlation between the error $|\llbracket \frac{\partial Sg_n}{\partial n} \rrbracket - g_n|$ and the $L^2(\Omega)$ error.

This error does not show up in the original ESIC method as shown in Table 5.5.

The reason is that in the latter method, polynomial refinement is performed near the interface. In this example, \mathbb{P}_1 elements have been used except for the elements containing the interface where a \mathbb{P}_2 basis was defined. As shown at the beginning of this section, when \mathbb{P}_2 are used, the error $|\llbracket \frac{\partial Sg_n}{\partial n} \rrbracket - g_n|$ has a second order convergence and then has the same behavior as the interpolation error. However this approach

TABLE 5.5
Convergence rates for 1D problem using the ESIC method.

Norm	Convergence rate
$L^2(\Omega)$	2
$H^1(\Omega)$	1
$L^\infty(\Omega)$	2

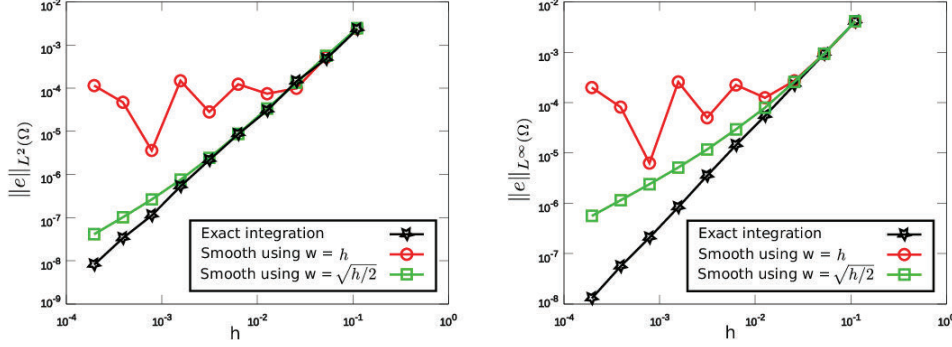


FIG. 5.3. Broken $L^2(\Omega)$ error (left) and $L^\infty(\Omega)$ error (right) associated with the different integration methods.

requires an additional programming effort as well as unnecessary addition of degrees of freedom: Figure 5.1 (bottom right) shows the pointwise error for the ESIC method and we can observe that the error in the element containing Γ is smaller than in the rest of the domain.

5.1.4. Solution based on smoothed integration. Finally, we investigate the effects of computing integrals using the regularized integrands introduced in section 3.2. We test both widths $w = \sqrt{h/2}$ and $w = h$. The effects of the thickness of the regularization band is clearly visible in Figure 5.3 (left) where we show the behavior of the broken $L^2(\Omega)$ norm of the error.

As stated previously in this section, optimal order of convergence is achieved with the exact integration. Using smooth integration with w proportional to \sqrt{h} , we can guarantee an almost optimal convergence in L^2 norm, while the choice $w = h$ leads to an unpredictable behavior when h becomes small. The convergence rate in that case is difficult to assess. We can observe the same kind of behavior in the $L^\infty(\Omega)$ norm as shown in Figure 5.3 (right). In this case we can see that using w proportional to \sqrt{h} leads to a convergence slower than the optimal one: if for coarse meshes the convergence rate seems to be close to 2, it then slows down to 1 for finer meshes.

All these results correspond quite well to the remarks that we made in section 3.2. The smooth integration using w proportional to \sqrt{h} permits to control the error leading to regular convergence. This means that the error generated near the interface, reported in the $L^\infty(\Omega)$ norm, is confined in that area and does not pollute the solution in the whole domain. On the contrary, with $w = h$, we lose the control on the quadrature error causing a large error in the interface area that eventually spreads in the whole domain.

5.2. 2D test case. We present here three test cases in $\Omega = (-1, 1)^2$, taking interfaces with different regularities. For all the tests performed, the integration procedure described in section 3.2 is used. We measure the error in the following

TABLE 5.6
Error for different meshes for the SESIC applied to the problem of section 5.2.1.

n	Error $L^2(\Omega)$	Rate	Error $H^1(\Omega)$	Rate	Error l^∞	Rate	Error $l^\infty(\Gamma)$	Rate
9	2.90e-02		4.10e-01		3.68e-02		5.72e-02	
19	6.07e-03	2.09	1.72e-01	1.16	8.13e-03	2.02	1.14e-02	2.16
39	1.36e-03	2.08	8.26e-02	1.02	1.83e-03	2.08	3.55e-03	1.62
79	2.57e-04	2.36	4.06e-02	1.01	2.89e-04	2.61	4.30e-04	2.99
159	5.67e-05	2.16	2.01e-02	1.00	1.24e-04	1.21	2.14e-04	0.99

norms:

- the broken H^1 norm of the error in the domain Ω ;
- the broken L^2 norm of the error in the domain Ω ;
- what we denote by l^∞ norm of the error, which is the maximal error in the finite element nodes;
- the $L^\infty(\Gamma)$ norm, used for the sake of comparison with other methods.

5.2.1. C^∞ interface. We first test our method on the 2D test case defined in [20]. This test is quite simple as the exact solution is continuous, so that the jump is only in the normal derivative. The domain is defined as the square $\Omega = (-1, 1)^2$ and the interface is the circle with radius 0.5 centered at the origin.

The exact solution reads

$$u(x, y) = \begin{cases} 1 & \text{if } x^2 + y^2 \leq 0.25, \\ 1 - \log(2\sqrt{x^2 + y^2}) & \text{if } x^2 + y^2 > 0.25. \end{cases}$$

Dirichlet boundary conditions are set to ensure this exact solution and the jump in the normal derivative to be $\left[\frac{\partial u}{\partial n}\right]_\Gamma = -2$.

Cartesian meshes with n cells on each side were used. The results that we obtained are listed in Table 5.6.

The SESIC method yields optimal orders of convergence both at the interface and at the finite element nodes. This means that the error decreases with the same rate everywhere in the domain, including the neighborhood of the interface and the interface itself. Moreover, the magnitude of the error is comparable with those obtained with the methods (ESIC, XFEM, and IBM) compared in [20], while being easier to implement and cheaper to compute. For this test case, the SESIC method also yields optimal orders of convergence in the $L^2(\Omega)$ and $H^1(\Omega)$ norms.

5.2.2. C^1 interface. We investigate now the influence of the regularity of the interface. The next test case consists in a domain cut by a C^1 curve: the level set function is defined as (see Figure 5.4)

$$\phi(x, y) = \begin{cases} \sqrt{x^2 + (y - 0.5)^2} - 0.2 & \text{if } y > 0.5, \\ |x| - 0.2 & \text{if } |y| \leq 0.5, \\ \sqrt{x^2 + (y + 0.5)^2} - 0.2 & \text{if } y < -0.5 \end{cases}$$

and the exact solution is defined as

$$(5.4) \quad u(x, y) = \begin{cases} 1 - \log(2\sqrt{x^2 + y^2}) & \text{if } \phi(x, y) \geq 0, \\ 0 & \text{if } \phi(x, y) < 0. \end{cases}$$

The Dirichlet boundary conditions on $\partial\Omega$ and the jump conditions across Γ are computed using this exact solution. The errors obtained with the SESIC method are

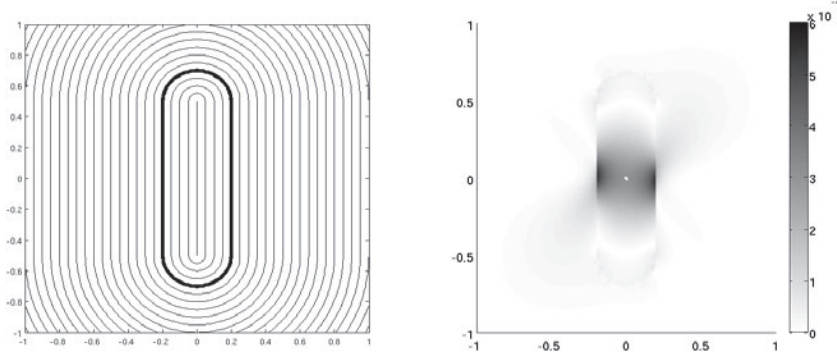


FIG. 5.4. Representation of the level set function (left, Γ bold) and pointwise error produced by the SESIC method with $n = 99$ (right).

TABLE 5.7
Error for different meshes for the SESIC applied to the problem from section 5.2.2.

n	$H^1(\Omega)$ error	Rate	$L^2(\Omega)$ error	Rate	$L^\infty(\Omega)$ error	Rate
9	6.51e-01		4.49e-02		9.07e-02	
19	3.20e-01	0.95	1.19e-02	1.78	3.50e-02	1.27
39	1.56e-01	1.00	3.21e-03	1.82	1.09e-02	1.62
79	7.71e-02	0.99	8.83e-04	1.83	3.24e-03	1.73
139	3.84e-02	0.99	2.65e-04	1.72	1.02e-03	1.65

TABLE 5.8
Error for different meshes for the SESIC applied to the problem from section 5.2.3.

n	$H^1(\Omega)$ error	Rate	$L^2(\Omega)$ error	Rate	$L^\infty(\Omega)$ error	Rate
9	3.04e-01		5.30e-02		7.74e-02	
19	1.57e-01	0.88	2.73e-02	0.89	5.13e-02	0.55
39	7.89e-02	0.96	1.41e-02	0.92	2.87e-02	0.81
79	4.08e-02	0.93	7.12e-03	0.97	1.65e-02	0.78
139	2.45e-02	0.91	4.07e-03	0.99	1.09e-02	0.73

reported in Table 5.7. We can remark that the convergence rate in H^1 is optimal whereas the ones in L^2 and L^∞ norms are between 1.5 and 2.

5.2.3. C^0 interface. Finally, we use the SESIC method to approximate the solution of a problem where Γ is only C^0 . We define the level set function as

$$\phi(x, y) = \begin{cases} \sqrt{(x - \frac{\sqrt{2}}{4})^2 + y^2} - 0.5 & \text{if } x \geq 0, \\ \sqrt{(x + \frac{\sqrt{2}}{4})^2 + y^2} - 0.5 & \text{if } x < 0 \end{cases}$$

and the exact solution is defined as in (5.4). Table 5.8 reports the errors obtained in this test case.

The lack of regularity of Γ is directly reflected in the convergence orders: the H^1 error as well as those in L^2 and L^∞ norm do not exceed the first order convergence. This can also be seen when looking at the pointwise error (Figure 5.5) where one can remark that the largest errors are created near the two points of low regularity of Γ .

5.3. 3D test case. We finally consider a 3D problem. This example was implemented in the parallel version of the finite element library LifeV (www.lifev.org).

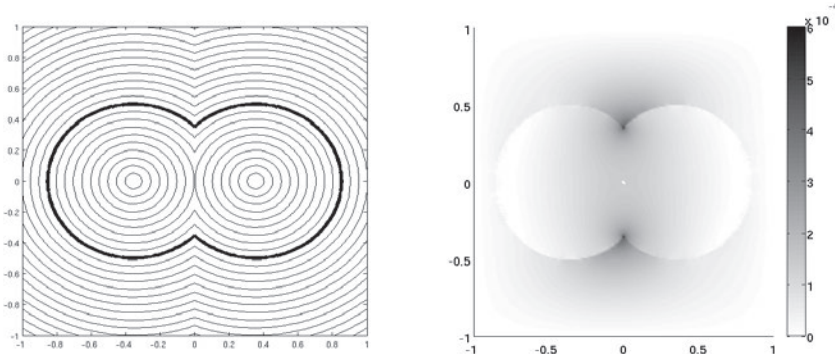


FIG. 5.5. Representation of the level set function (left, Γ bold) and pointwise error produced by the SESIC method with $n = 99$ (right).

TABLE 5.9

Error for different meshes for the SESIC applied to the 3D problem, with integration width $w = 0.5\sqrt{h}$.

n	Error $L^2(\Omega)$	Rate	Error $H^1(\Omega)$	Rate	Error l^∞	Rate
9	1.61e-01		1.86e-00		1.19e-01	
19	2.25e-02	2.63	4.41e-01	1.93	3.14e-02	1.78
39	2.84e-03	2.88	1.01e-01	2.05	8.59e-03	1.80
59	8.63e-03	2.88	4.34e-02	2.04	4.10e-03	1.79
79	4.01e-04	2.63	2.41e-02	2.02	2.46e-03	1.75
99	1.90e-04	3.31	1.52e-02	2.04	1.66e-03	1.74
119	1.20e-04	2.50	1.05e-02	2.01	1.22e-03	1.67

We consider the domain $\Omega = (-1, 1)^3$ with $\phi(x, y, z) = \sqrt{x^2 + y^2 + z^2} - 0.5$ so that the interface Γ is a sphere centered on the origin with a radius 0.5. In Ω , we want to find the solution $u : \Omega \rightarrow \mathbb{R}$ of the problem $-\Delta u = 0$ with jump conditions through Γ : $\llbracket u \rrbracket_\Gamma = 2 - e^{x+z} \sin(\sqrt{2}y)$ and

$$\left[\frac{\partial u}{\partial n} \right]_\Gamma = 4 + 2e^{x+z} \left((x+z) \sin(\sqrt{2}y) + \sqrt{2}y \cos(\sqrt{2}y) \right).$$

The boundary conditions are such that the exact solution is

$$u(x, y, z) = \begin{cases} (x^2 + y^2 + z^2)^{-1/2} & \text{if } \phi(x, y, z) \geq 0, \\ e^{x+z} \sin(\sqrt{2}y) & \text{if } \phi(x, y, z) < 0. \end{cases}$$

We solved this problem using \mathbb{P}_1 finite elements. To measure the error, we computed both the broken L^2 and H^1 error in the domain Ω and maximum error in the finite element nodes (denoted hereafter by l^∞ as in the 2D case). We used smoothed integration on Cartesian meshes with n being the number of subdivisions in each direction. The computed errors and convergence rates are given in Table 5.9.

The convergence rate in the broken $L^2(\Omega)$ norm is slightly higher than the optimal rate 2. The same remark holds for the broken $H^1(\Omega)$ norm, where the convergence looks close to 2. Finally, the convergence rate is between 1.5 and 2 in l^∞ norm. The pointwise error computed on the surface $x = 0$ is shown in Figure 5.6.

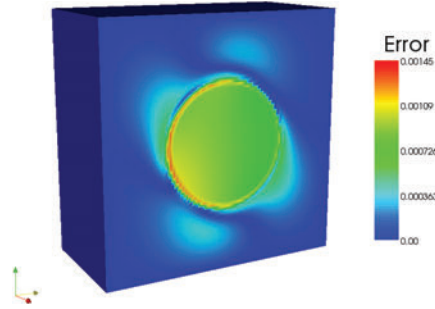


FIG. 5.6. Representation of the error on the surface $x = 0$ for $n = 80$. We can remark that the error near the interface is of the same order of magnitude as far from it.

6. Application to free surface flows. We present an application of the SESIC method to free surface flows, referring to [14, 32] for more details.

We consider the problem of simulating the motion of a surface separating two incompressible Newtonian fluids. Differently from the problems in the previous sections, the position of the interface is now unknown and described through a level set function. To represent the motion of the two fluids, we use the Navier–Stokes equations on each side of the surface. The density ρ and the viscosity μ of the fluid are constant within each subregion, but they are discontinuous across the interface where they can differ by several orders of magnitude. For example, in the case of water and air, the density ratio is approximately 1000 whereas the viscosity ratio is 50. Due to the difference of densities, the gravity force $\rho\mathbf{g}$ exhibits a large discontinuity across the interface that leads to a jump in the normal derivative of the pressure. At the discrete level, a standard choice of the finite element spaces does not allow us to represent discontinuities in the gradients of the unknown pressure inside the elements crossed by the free surface. This can result in oscillations in the pressure that may possibly pollute the whole simulation and give an unphysical shape to the surface (this is especially visible when the viscosities are low).

This problem had already been studied in [11] where a pressure correction to get rid of these unphysical oscillations is proposed. More precisely, new finite element shape functions with discontinuous gradients across the interface are added and the new degrees of freedom are condensed a priori. The main disadvantages of this approach are the construction of the new shape functions done geometrically in each element and the extra cost represented by the a priori condensation.

Using the SESIC method, we can provide a simpler and faster way of constructing such a pressure correction. Indeed, since the jump in the gradient of the pressure is known a priori,

$$\left[\left[\frac{\partial p}{\partial n} \right] \right]_{\Gamma} = \llbracket \rho \rrbracket_{\Gamma} |\mathbf{g}|,$$

we can construct a lifting for this jump like (3.9) with $g_n = \llbracket \rho \rrbracket_{\Gamma} |\mathbf{g}|$. Then, we subtract this lifting from the original pressure obtaining an additional term in the right-hand side of the momentum equation of the Navier–Stokes system. The lifting can be built without reconstructing the interface as explained in the previous sections.

6.1. Numerical results. We consider a cylindrical vessel of radius 0.144 m filled with water up to height 0.15 m. The cylinder undergoes a rotational movement

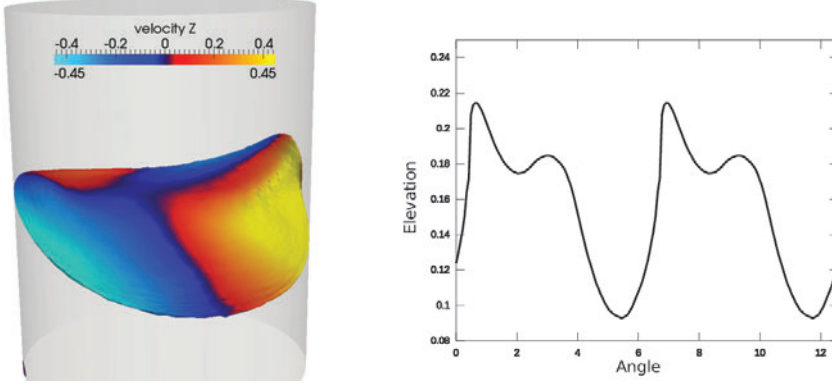


FIG. 6.1. (Left) Shape of the free surface at 80 rpm (colored by vertical velocity) and (right) its trace on the lateral wall for different values of a .

around an axis not coinciding with its own axis, the distance between the center of rotation and the center of the cylinder being equal to 0.025 m. The agitation rate is $\omega = 80$ rpm, corresponding to Froude number $Fr = 1.4357$. Numerical simulation can capture correctly the free surface generated at the steady state; according to experimental results, they show that the surface wave has two main peaks with one of them that can break. The shape of the free surface together with the magnitude of the vertical velocity are shown in Figure 6.1 (left). On the right-hand side of the same figure we show the trace of the free surface on the later boundary at different angles $\theta_a = a\pi/45$, $a = 0, \dots, 12$.

7. Conclusions. In this paper, we have investigated a new method, the SESIC method, that can be used to solve interior discontinuity interface problems. It relies on simple construction of liftings, i.e., finite element functions that are built to carry the discontinuities across the interface.

The method that we proposed was inspired by the ESIC method (see [20]) and in fact inherits some properties of that method. First of all, the jumps across the interface are actually reproduced by the method. The cost for this method is also quite low, as only the assembly for a small part (corresponding to the elements crossed by the interface) of the right-hand side is needed.

The SESIC method has also the advantage of requiring no additional degree of freedom, as no new basis function nor refinement near the interface are necessary. The consequence is that the stiffness matrix remains unchanged with respect to that associated with the given PDE without interface discontinuities. At the algebraic level, solution strategies and preconditioning need not be modified. The liftings introduced in the SESIC method have removed one of the bottlenecks of the ESIC method: there is no need to reconstruct the interface explicitly for building the liftings. This adds more generality to the method, as the level set can now be given in all the possible forms: even a level set given as a finite element function with high polynomial degree would be acceptable, as there is no need to solve a nonlinear equation for finding the zero level set. Moreover, if we use regularizing functions for the integration of discontinuous integrands, then the SESIC method treats the interface in a fully implicit way.

The numerical results show that in the 1D and 2D cases that we have tested, the method exhibits optimal orders of convergence. The use of regularized integrands

rather than exact integration across the interface leads to a slower convergence in the regularization band around the interface, but it keeps the optimal convergence in the remaining part of the domain, as shown on the 3D test case that we have considered.

Acknowledgments. We would like to acknowledge professors J. Sethian and J.-S. Huh for many useful comments on an earlier version of this paper.

REFERENCES

- [1] P. AREIAS AND T. BELYTSCHKO, *Analysis of three-dimensional crack initiation and propagation using the extended finite element method*, Internat. J. Numer. Methods Engrg., 63 (2005), pp. 760–788.
- [2] E. BÉCHET, H. MINNEBO, N. MOËS, AND B. BURGARDT, *Improved implementation and robustness study of the X-FEM for stress analysis around cracks*, Internat. J. Numer. Methods Engrg., 64 (2005), pp. 1033–1056.
- [3] J. BEDROSSIAN, J. VON BRECHT, S. ZHU, E. SIFAKIS, AND J. M. TERAN, *A second order virtual node method for elliptic problems with interfaces and irregular domains*, J. Comput. Phys., 229 (2010), pp. 6405–6426.
- [4] T. BELYTSCHKO, N. MOES, S. USUI, AND C. PARIMI, *Arbitrary discontinuities in finite elements*, Internat. J. Numer. Methods Engrg., 50 (2001), pp. 993–1013.
- [5] E. BURMAN AND P. HANSBO, *Fictitious domain finite element methods using cut elements: II. A stabilized Nitsche method*, Appl. Numer. Math., 62 (2012), pp. 328–341.
- [6] E. BURMAN, *Ghost penalty*, C. R. Math., 348 (2010), pp. 1217–1220.
- [7] G. CAGINALP AND X. CHEN, *Phase field equations in the singular limit of sharp interface problems*, in On the Evolution of Phase Boundaries, IMA Vol. Math. Appl. 43, Springer, New York, 1992, pp. 1–27.
- [8] K. W. CHENG AND T.-P. FRIES, *Higher-order XFEM for curved strong and weak discontinuities*, Internat. J. Numer. Methods Engrg., 82 (2010), pp. 564–590.
- [9] J. CHESSA, P. SMOLINSKI, AND T. BELYTSCHKO, *The extended finite element method (XFEM) for solidification problems*, Internat. J. Numer. Methods Engrg., 53 (2002), pp. 1959–1977.
- [10] D. L. CHOPP, *Some improvements of the fast marching method*, SIAM J. Sci. Comput., 23 (2001), pp. 230–244.
- [11] A. COPPOLA-OWEN AND R. CODINA, *Improving Eulerian two-phase flow finite element approximation with discontinuous gradient pressure shape functions*, Internat. J. Numer. Methods Fluids, 49 (2005), pp. 1287–1304.
- [12] J. COTTRELL, T. HUGHES, AND Y. BAZILEVS, *Isogeometric Analysis: Toward Integration of CAD and FEA*, John Wiley, Hoboken, NJ, 2009.
- [13] M. DISCACCIATI, D. HACKER, A. QUARTERONI, S. QUINODOZ, S. TISSOT, AND F. WURM, *Numerical simulation of orbitally shaken viscous fluids with free surface*, Int. J. Numer. Methods Fluids, 71 (2013), pp. 294–315.
- [14] M. DISCACCIATI, D. HACKER, A. QUARTERONI, S. QUINODOZ, S. TISSOT, AND F. WURM, *Numerical simulation of orbitally shaken viscous fluids with free surface*, Internat. J. Numer. Methods Fluids, 71 (2013), pp. 294–315.
- [15] D. DUNAVANT, *High degree efficient symmetrical Gaussian quadrature rules for the triangle*, Internat. J. Numer. Methods Engrg., 21 (1985), pp. 1129–1148.
- [16] A. ERN AND J.-L. GUERMOND, *Theory and Practice of Finite Elements*, Appl. Math. Sci. 159, Springer-Verlag, New York, 2004.
- [17] S. GROSS AND A. REUSKEN, *An extended pressure finite element space for two-phase incompressible flows with surface tension*, J. Comput. Phys., 224 (2007), pp. 40–58.
- [18] P. HANSBO, *Nitsche’s method for interface problems in computational mechanics*, GAMM-Mitt Ges. Angew. Math. Mech., 28 (2005), pp. 183–206.
- [19] X. HE, T. LIN, AND Y. LIN, *Immersed finite element methods for elliptic interface problems with non-homogeneous jump conditions*, Internat. J. Numer. Anal. Model., 8 (2010), pp. 284–301.
- [20] J.-S. HUH AND J. SETHIAN, *Exact subgrid interface correction schemes for elliptic interface problems*, Proc. Natl. Acad. Sci. USA, 105 (2008), pp. 9874–9879.
- [21] J.-S. HUH AND J. SETHIAN, *private communication*, Department of Mathematics, University of California, Berkeley, 2010.
- [22] J.-L. LIONS AND E. MAGENES, *Problèmes aux Limites Non Homogènes et Applications*, Vol. 1, Dunod, Paris, 1968.

- [23] J. LIONS AND E. MAGENES, *Non-Homogeneous Boundary Value Problems and Applications*, Springer, Berlin, 1972.
- [24] Z. LI AND M.-C. LAI, *The immersed interface method for the Navier-Stokes equations with singular forces*, J. Comput. Phys., 171 (2001), pp. 822–842.
- [25] D. NGUYEN, R. FEDKIW, AND M. KANG, *A boundary condition capturing method for incompressible flame discontinuities*, J. Comput. Phys., 172 (2000), pp. 71–98.
- [26] S. OSHER AND R. FEDKIW, *Level Set Methods and Dynamic Implicit Surfaces*, Springer, New York, 2002.
- [27] N. PAROLINI, *Computational Fluid Dynamics for Naval Engineering Problems*, Ph.D. thesis, Ecole Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, 2004.
- [28] D. PENG, B. MERRIMAN, S. OSHER, H. ZHAO, AND M. KANG, *A PDE-based fast local level set method*, J. Comput. Phys., 115 (1999), pp. 410–438.
- [29] A. QUARTERONI, R. SACCO, AND F. SALERI, *Numerical Mathematics*, Springer, Berlin, 2007.
- [30] A. QUARTERONI AND A. VALLI, *Numerical Approximation of Partial Differential Equations*, Springer, Berlin, 1994.
- [31] A. QUARTERONI, *Numerical Models for Differential Problems*, Springer, Berlin, 2009.
- [32] S. QUINODOZ, *Numerical Simulation of Orbitally Shaken Reactors*, Ph.D. thesis, Ecole Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, 2012.
- [33] J. SETHIAN, *Level Set Methods and Fast Marching Methods*, 2nd ed., Cambridge University Press, Cambridge, 1999.
- [34] A. TORNBERG, *Multi-dimensional quadrature of singular and discontinuous functions*, BIT, 42 (2002), pp. 644–669.
- [35] G. ZI AND T. BELYTSCHKO, *New crack-tip elements for XFEM and applications to cohesive cracks*, Internat. J. Numer. Methods Engrg., 57 (2003), pp. 2221–2240.